Remarks

The above Amendments and these Remarks are in reply to the Office Action mailed September 19, 2006 and the Examiner Interview of November 15, 2006. Applicant acknowledges with thanks Examiner Nguyen's assistance in granting an interview on November 15, 2006, during the course of which interview various features of the claimed embodiments were discussed, the substance of which is included herein.

Claims 1-22 were pending in the Application prior to the outstanding Office Action. In the Office Action, the Examiner rejected claims 1-22. The present Reply amends claims 1, 3, 5, 6, 7, and 14, and adds new claims 23 and 24, leaving for the Examiner's present consideration claims 1-24. Reconsideration of the rejections is requested.

Objections

Claim 14 was objected to due to containing an incorrect claim dependency.

Rejections under 35 U.S.C. §102

Claims 1-9, 14-17, and 19-22 were rejected under 35 U.S.C. §102(b) as being anticipated by Stapp et al. (U.S. Application No. 09/866,131).

Rejections under 35 U.S.C. §103

Claims 10-13 and 18 were rejected under 35 U.S.C. §103(a) as being unpatentable over Stapp in view of Goodwin (U.S. Pat. No. 6,199,195).

Response to Rejections

Claim 1 (as amended) states:

A computer program product including a storage medium with instructions thereon for execution by a computer for dynamically generating program code, the instructions comprising:

computer code for dynamically generating program code, wherein dynamically generating program code includes:

computer code for creating a class file container object;

computer code for adding a method to the class file object;

computer code for adding code to the method using programming language constructs;

computer code for generating byte code for the class file container object; and

computer code for instantiating an instance of the new class file object.

Applicant claims computer code for dynamically generating code. Programming language constructs are used to add code to the methods. The advantage of dynamic code generation is that it occurs at run-time (also known as execution-time), whereas traditional code generation occurs at development-time. Paragraph 22 of the specification gives a pseudo code example that uses the dynamic code generation described in claim 1 to produce the class described in paragraph 24 at run-time. The class of paragraph 24 is then instantiated and executed at run-time.

Stapp (U.S. Application No. 09/866,131) discloses code generation using XSLT. Stapp discloses code generation, but Stapp does not disclose dynamic code generation. Furthermore, Stapp discloses code generation using XSLT, but Stapp does not disclose using programming language constructs to add code to methods.

Dependent Claims 2-24 are believed to be patentable for reasons similar to those discussed above with Claim 1. It is also submitted that Claims 2-24 also add their own limitations which render them patentable in their own right. Applicant reserves the right to argue these limitations should it become necessary in the future.

Conclusion

In light of the above, it is respectfully submitted that all of the claims now pending in the subject patent application should be allowable, and a Notice of Allowance is requested. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: November 27, 2006

By: ______________________
Thomas K. Plunkett
Reg. No. 57,253

Customer No. 23910
FLIESLER MEYER LLP
Four Embarcadero Center, Fourth Floor
San Francisco, California 94111-4156
Telephone: (415) 362-3800
Facsimile: (415) 362-2928